Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A method in a data processing system for generating coverage data for accesses to dynamically allocated data during execution of code in a data processing system, the method comprising:

responsive to a request to dynamically allocate a memory area for dynamically allocated data during runtime when an allocation of memory is required, dynamically allocating the memory area;

responsive to dynamically allocating the memory area, associating the memory area with a data access indicator;

responsive to executing an instruction in the code at a processor in the data processing system, determining whether an access to a memory location associated with the data access indicator has occurred; and

if the data access indicator is associated with the memory area, changing a state of the data access indicator by the processor when the instruction is executed, wherein the coverage data for the dynamically allocated data is generated during execution of the code by the processor,

wherein the memory area comprises a starting memory location and an ending memory location in which the starting memory location and the ending memory location span a size of memory equal to the request, and a subsequent memory location located one byte after the ending memory location, wherein the data access indicator comprises a first data access indicator in a set of data access indicators associated with the memory area, and wherein the first data access indicator identifies the starting memory location, and wherein the associating step further comprises:

associating a second data access indicator in the set of data access indicators with the ending memory location, wherein the second data access indicator identifies the ending memory location in the memory area; and

associating a third data access indicator in the set of data access indicators with the subsequent memory location, wherein the third data access indicator identifies the subsequent memory location, wherein an access to the one byte after the ending memory location indicates that a memory size of the memory area is insufficient.

2. (Canceled)

3. (Canceled)

4. (Canceled)

5. (Original) The method of claim 1 further comprising:
retrieving call stack information in response to dynamically allocating the memory area.

6. (Original) The method of claim 5 further comprising:
identifying code making the request for the memory area using the call stack information.

7. (Original) The method of claim 5 further comprising:
determining calling sequences in the code using the call stack information.

8. (Original) The method of claim 1, wherein the access indicator is located in a field in the instruction.

9. (Original) The method of claim 1, wherein the access indicator associated with the instruction is located in a shadow memory.

10. (Original) The method of claim 1, wherein the access indicator associated with the instruction is located in a page table.

11. (Original) The method of claim 1, wherein memory location accessed during execution of the code have set data access indicators set when the state of access indicators associated with an executed instruction are changed, while memory location unaccessed during execution of the code have unset data access indicators because the state of the unset data access indicators remain unchanged.

12. (Currently amended) A data processing system for generating coverage data for accesses to dynamically allocated data during execution of code in a data processing system, comprising:

allocating means, responsive to a request to dynamically allocate a memory area for dynamically allocated data during runtime when an allocation of memory is required, for dynamically allocating the memory area;

associating means, responsive to dynamically allocating the memory area, for associating the memory area with a data access indicator;

determining means, responsive to executing an instruction in the code at a processor in the data processing system, for determining whether an access to a memory location associated with the data access indicator has occurred; and

changing means for changing a state of the data access indicator by the processor when the instruction is executed if the data access indicator is associated with the memory area, wherein the coverage data for the dynamically allocated data is generated during execution of the code by the processor, wherein the memory area comprises a starting memory location and an ending memory location in which the starting memory location and the ending memory location span a size of memory equal to the request, and a subsequent memory location located one byte after the ending memory location, wherein the data access indicator comprises a first data access indicator in a set of data access indicators associated with the memory area, wherein the first data access indicator identifies the starting memory location in the memory area, and wherein the associating means comprises first associating means for associating the first data access indicator in the set of data access indicators with the starting memory location, wherein the associating means further comprises:

second associating means for associating a second data access indicator in the set of data access indicators with the ending location, wherein the second data access indicator identifies the ending memory location in the memory area; and

third associating means for associating a third data access indicator in the set of data access indicators with the subsequent memory location, wherein the third data access indicator identifies the subsequent memory location, wherein an access to the one byte after the ending location indicates that a memory size of the memory area is insufficient.

13. (Canceled)

14. (Canceled)

15. (Canceled)

16. (Original) The data processing system of claim 12 further comprising:

retrieving means for retrieving call stack information in response to dynamically allocating the memory area.

17. (Original) The data processing system of claim 16 further comprising:

identifying means for identifying a process in code making the request for the memory area using the call stack information.

18. (Original) The data processing system of claim 16 further comprising:

determining means for determining calling sequences in the code using the call stack information.

19. (Original) The data processing system of claim 12, wherein the access indicator is located in a field in the instruction.

20. (Original) The data processing system of claim 12, wherein the access indicator associated with the instruction is located in a shadow memory.

21. (Original) The data processing system of claim 12, wherein the access indicator associated with the instruction is located in a page table.

22. (Original) The data processing system of claim 12, wherein memory location accessed during execution of the code have set data access indicators set when the state of access indicators associated with an executed instruction are changed, while memory location unaccessed during execution of the code have unset data access indicators because the state of the unset data access indicators remain unchanged.

23. (Currently amended) A computer program product in a recordable-type computer readable medium for generating coverage data for accesses to dynamically allocated data during execution of code in a data processing system, the computer program product comprising:

first instructions, responsive to a request to dynamically allocate a memory area for dynamically allocated data during runtime when an allocation of memory is required, for dynamically allocating the memory area;

second instructions, responsive to dynamically allocating the memory area, for associating the memory area with a data access indicator;

third instructions, responsive to executing an instruction in the code at a processor in the data processing system, for determining whether an access to a memory location associated with the data access indicator has occurred; and

fourth instructions for changing a state of the data access indicator by the processor when the instruction is executed if the data access indicator is associated with the memory area, wherein the coverage data for the dynamically allocated data is generated during execution of the code by the processor, wherein the memory area comprises a starting memory location and an ending memory location in which the starting memory location and the ending memory location span a size of memory equal to the request, and a subsequent memory location located one byte after the ending memory location, wherein the data access indicator comprises a first data access indicator in a set of data access indicators associated with the memory area, wherein the first data access indicator identifies the starting memory location, and wherein the second instructions comprise first sub-instructions for associating the first data access indicator with the starting memory location, and wherein the second instructions further comprises:

second sub-instructions for associating a second data access indicator in the set of data access indicators with the ending location, wherein the second data access indicator identifies the ending memory location in the memory area; and

third sub-instructions for associating a third data access indicator in the set of data access indicators with the subsequent memory location, wherein the third data access indicator identifies the subsequent memory location, wherein an access to the one byte after the ending location indicates that a memory size of the memory area is insufficient.

24. (Canceled)

25. (Original) The computer program product of claim 23 further comprising:

fifth instructions for retrieving call stack information in response to dynamically allocating the memory area.

26. (Original) The computer program product of claim 25 further comprising:

sixth instructions for identifying a process in code making the request for the memory area using the call stack information.